FPGA IMPLEMENTATION OF FLOOD MONITORING SYSTEM

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIRMENTS FOR THE

DEGREE OF

MASTER OF TECHNOLOGY

IN

VLSI DESIGN & EMBEDDED SYSTEM

By

SUCHITAL DEBBARMA

Roll No: 212EC2133

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA, ODISHA

INDIA

2014

FPGA IMPLEMENTATION OF FLOOD MONITORING SYSTEM

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIRMENTS FOR THE

DEGREE OF

MASTER OF TECHNOLOGY

IN

VLSI DESIGN & EMBEDDED SYSTEM

By

SUCHITAL DEBBARMA

Roll No: 212EC2133

UNDER THE GUIDANCE OF

Prof. DEBIPRASAD PRIYABRATA ACHARYA

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA, ODISHA

INDIA

2014

NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

CERTIFICATE

This is to certify that the thesis entitled, **"FPGA IMPLEMENTATION OF FLOOD MONITORING SYSTEM"** submitted by **SUCHITAL DEBBARMA** in partial fulfilment of the requirements for the award of Master of Technology Degree in **Electronics & Communication Engineering** with specialization in **VLSI DESIGN & EMBEDDED SYSTEM** during 2013-2014 at the National Institute of Technology, Rourkela (Deemed University) is a study work carried out by him under my supervision and guidance. To the best of my knowledge, the matter embodied in the thesis has not been submitted by him to any other University / Institute for the award of any Degree or Diploma.

DATE

Prof. D.P ACHARYA
(Supervisor)
Dept. Of Electronics & Communication Engg.
National Institute of Technology
Rourkela-769008

ACKNOWLEDGEMENT

First of all I would like to take this opportunity to heartily express my thanks to my supervisor for the project **Prof. Debiprasad P. Acharya** for helping me out and providing valuable insight to the project work throughout the completion of the project despite his hectic hour in the department.

I would also like to thank **Prof. Ayas K. Swain** for helping me complete the work. And also I would like to thank **Mr. P.k Sahoo**, Sr. Technical assistant, **Mr. Jaganath Mahanty**, **Mr. G.Tom Varghese**, and **Mr. Sudeendra**, Phd, ECE department for providing me the necessary hardware components required to carry the work.

I would like to thank all the faculty members of ECE department for their guidance and helping hands.

Finally I would like to thank all my friends for always giving me a helping hand in my difficulty times.

Last but not the least I would like to thank my parents and the Almighty God for giving me all the opportunity in my life.

Suchital Debbarma

ABSTRACT

Flood is one of the huge disasters in India which affects the human, animals, fertile agricultural land, soil etc. It is not possible to fully control the flood but we can definitely reduce the damage done by flood & its impact on human & animal life. In this paper we explain flood prediction system using sensor and processing in FPGA board. The sensor is used to measure real world parameters like water pressure, water level, flow, temperature etc. The output of sensor is then given to FPGA board to convert the physical data (AC or DC voltage) to digital data to do further processing on the board. This system is implemented in the river in such a way that the sensor will able to detect the water level at any time. The main motive behind the design of the system are indigenous system, low cost, low power consumption, easy installation process, highly reliable and people living the adjoining flood prone area can be given early warning of the disaster. Predicting the flood before its actual occurrence can buy sufficient time for residents to evacuate nearby areas, preventing loss of life and property. This project will be Helpful throughout the year. The design has been prototyped on Xilinx Spartan 3E FPGA starter kit board.

CONTENTS

LIST OF FIGURES

LISTS OF TABLES

LISTS OF ACRONYMS

ADC	Analog-to-Digital Converter
AD_CONV	ADC Convert
AMP	Amplifier
AMP_CS	AMP_Chip Select
AMP_DOUT	AMP_Data out
AMP_SHDN	AMP_Shutdown
AT	Attention
DAC	Digital-to-Analog Converter
DCE	Data Communications Equipment
DTE	Data Terminal Equipment
FPGA	Field Programmable Gate Array
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
LCD	Liquid Crystal Display
LED	Light Emitting Diode
LVCMOS	Low-Voltage Complementary MOS
LVTTL	Low-Voltage Transistor-to-Transistor Logic
MSB	Most significant bit
PGA	Programmable Gain Amplifier
PS/2	Personal system/2
RS232	Recommended Standard 232
SPI	Serial Peripheral Interface
SPI_MOSI	SPI_Master Out Slave In
SPI_MISO	SPI_Master In Slave Out
SPI_SCK	SPI_System clock
TCP/IP	Transport Control Protocol/Internet Protocol Suite
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
VGA	Video Graphics Array

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION TO FLOOD MANAGEMENT SYSTEM

Flooding is one of the key disasters occurring in different region of the world. Flood has been one of the major problems in most of the states in India. Flooding happens after intense rains, when rivers overflow, when oceanic waves come back onto land, when snow melts fast or once dams overflow or break. Flooding is that the most typical of all natural hazards in Asian nation particularly throughout monsoon and causes important and ire-coverable harm to life and properties of the people.

Although, we may be able to predict rainfall or track rainstorm accurately via satellite images, we have to have real-time monitored information like flow, rainfall level, or water level etc. so that sensible action can be taken before hand to prevent flooding. In Orissa, flooding is a recurrent event affecting the entire state, especially the river bank and coastal areas. Every year, it causes lives and damages to infrastructure, agricultural production and severely affects local economic development. As a result, flood management of the state is a crucial challenge.

Disaster flood alert system using pressure sensors is one of the cheapest technology available today which is useful to make the people alert from disaster flood [1]. In this project pressure sensors are used to find out the water level of the river/dam. Every year disaster flood has its adverse affects. Due to this government have to face more critical problems. Many people with animals have to strive for their life. Overall it affects the ecological balance, crops diorites due to excess water and is carried away with it. Soil erosion takes place to large extent.

Flood monitoring using real-time sensor is one of the non-structural flood control measures. Losses due to flooding can be reduced by means of measures such as monitoring, forecasting, simulation, evaluation, and analysis. The effective implementation of flood monitoring and warning system is non-trivial, since it requires the reliability coupled with the

availability of related information. These papers provide great insights into the development of flood forecasting and alert system using pressure sensor.

The design system is composed of two parts: signal conditioning system and transmitting system. We employ water pressure sensor to measure the water level/flow.

1.2 OBJECTIVE

The main objective towards the work is to design a cheap and indigenous system that will monitor the water level of River or Dam so that the flood caused by the river/dam water can be reduced to a great extent and which will be a significant helpful for the mankind.

1.3 LITERATURE REVIEW

- Sandeep Gandla, Waleed K. Al-Assadi, Sahra Sedigh and Raghu A. R. Rao, "Design and FPGA Prototyping of a Flood Prediction System", Department of Electrical and Computer Engineering, Missouri University of Science and Technology [1].

This paper depicts system design of flood monitoring system using piezoelectric pressure sensor. The utmost criterion of the paper include low cost, low power consumption, reliability etc.

- Inyiama H. C., Obota M. E.,(2013), "Designing Flood Control Systems Using Wireless Sensor Networks", International Journal of Engineering Research and Applications (IJERA), Vol. 3, Issue 1, January -February 2013, pp.1374-1382 [2].

This paper describes an automatic flood alert system based on supervised control. The design uses water level sensor and wireless sensor network which automatically control flood.

- Jirapon Sunkpho, Chaiwat Ootamakorn,(2011), Real-time flood monitoring and warning system, Songklanakarin Journal of Science and Technology, 33 (2), 227-235, Mar. - Apr. 2011 [3].

In this paper real-time data of water condition is monitored by using wireless sensors network Utilizing mobile General Packet Radio Service (GPRS) communication to transmit measured data to the Control room.

CHAPTER 2

SYSTEM DESIGN ARCHITECTURE

2.1 INTRODUCTION TO SYSTEM ARCHITECTURE

The system architecture of the system design is given in figure 1. The flood parameters like water pressure, water level, flow, temperature etc are measured by the sensor [1]. The output voltage from the sensor is given to the onboard ADC present on the Spartan 3e FPGA board used in the project.

2.2 MAJOR COMPONENTS BLOCK

The different components or block of the system design are:-

1. Sensors/Transducers
2. Pre-amplifier & ADC interfacing
3. RS-232 transmitter interfacing
4. LCD interfacing

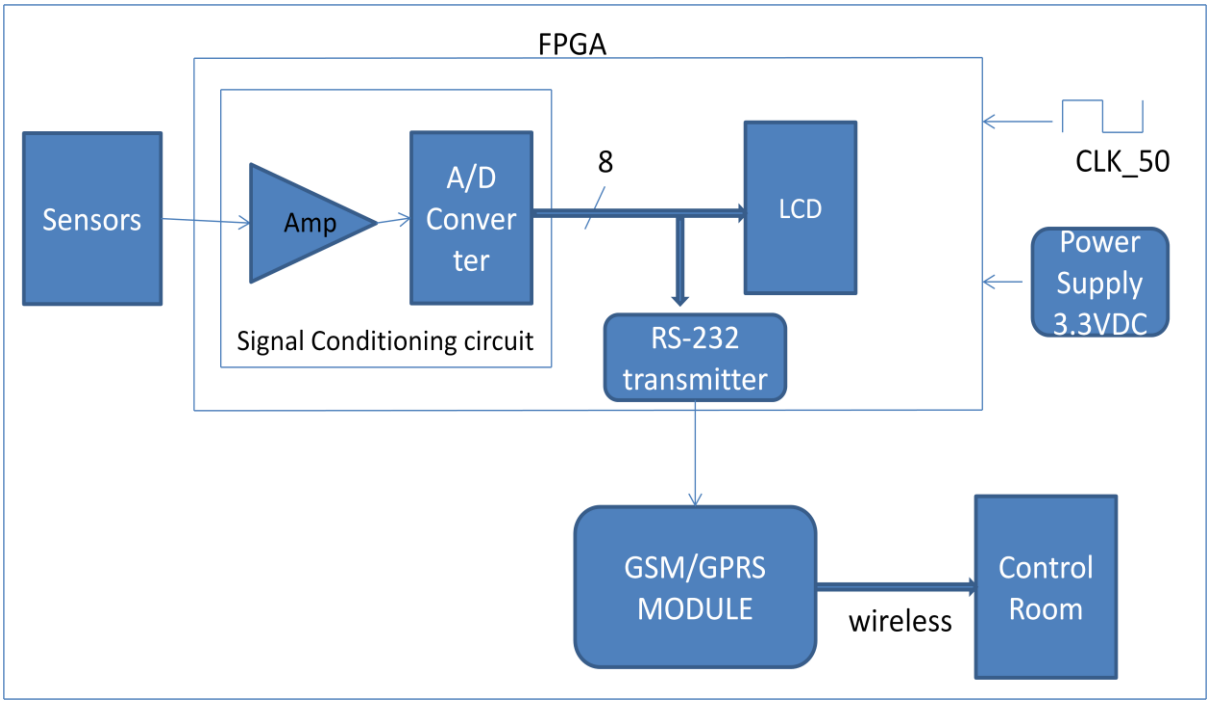

Figure 1 Block diagram of the system design

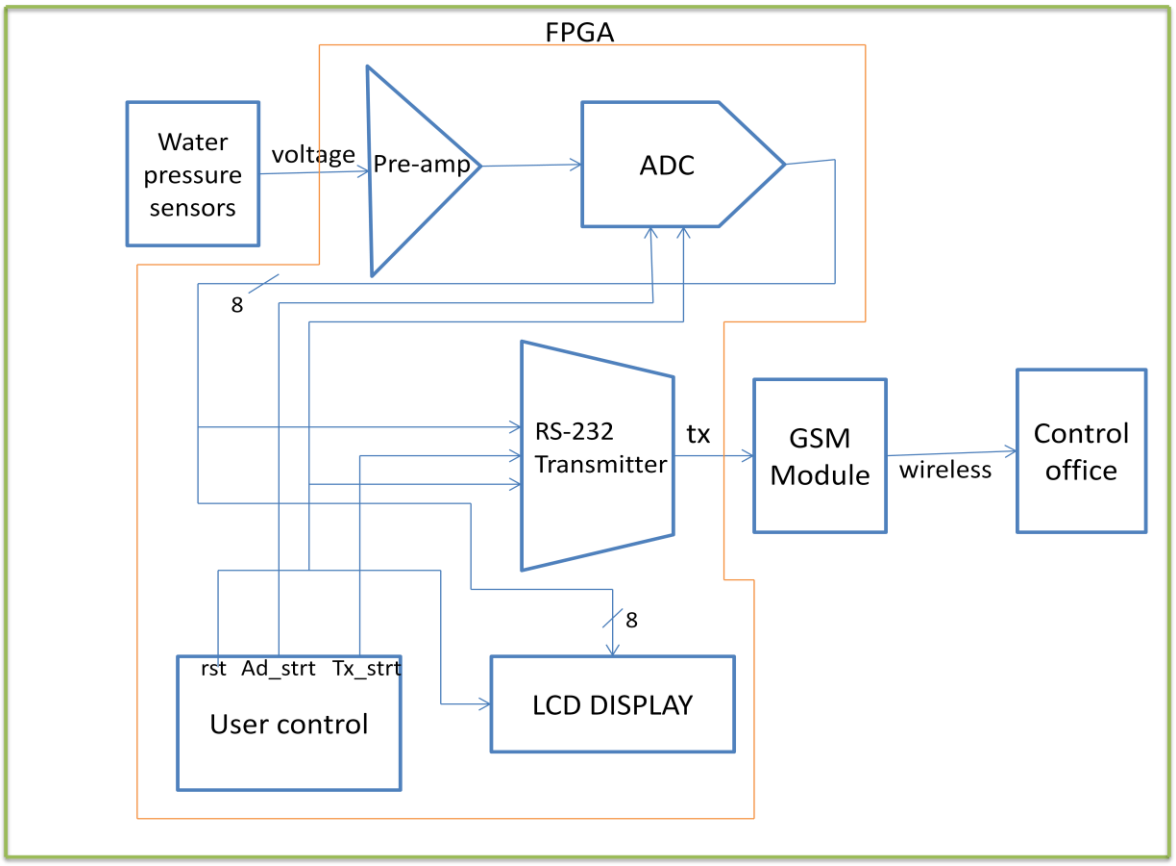

Figure 2 Internal Architecture of the system design

CHAPTER 3

PRE-AMPLIFIER & ADC INTERFACING

3.1 INTRODUCTION TO PRE-AMPLIFIER PRESENT ON FPGA BOARD

There are 2 separate inverting Amplifier on the Spartan 3e FPGA board[7]. The incoming analog signal is amplified proportional to 1.65V. Gain can be programmed between -1 and -100. It is a 2 channel gain amplifiers. For both the channels gains are programmed in parallel via 3-wire SPI internal bus for selecting gains of 0, -1, -2, -5, -10, -20, -50 and -100.

3.1.1 PRE-AMPLIFIER INTERFACING WITH FPGA

FPGA and pre-amplifier are interfaced with the help of various signals. SPI bus is used for communication between pre-amplifier and the FPGA. SPI bus is also shared between ADC, DAC, Platform Flash and Strata Flash[7]. So when one device is interfacing with SPI bus other device should be disable to enable proper interface between the ongoing connection. SPI signals which control proper connection of pre-amplifier and FPGA are :-

- **SPI_MOSI:** It is an internal bus with pin T4 on the FPGA board. It is the output signal to amplifier. This bus gives serial data. It gives programmable gain byte[7].
- **AMP_CS:** SPI bus with pin N7 on the FPGA board. Directed from FPGA to amplifier. Active-low chip select signal. When the signal returns high the amplifier gain is set[7].
- **SPI_SCK: P**in U16 on the FPGA board. Directed from FPGA to amplifier. Clock signal which sets the gain[7].
- **AMP_SHDN: P**in P7 on FPGA board. FPGA to the amplifier.
- **AMP_DOUT:** Pin E18 on the FPGA board. Directed from AMP to FPGA. Echoes earlier AMP gain data back to FPGA.

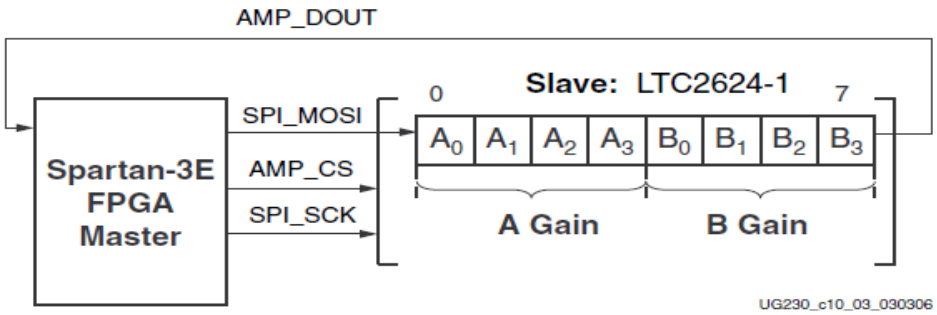

Figure 3 SPI Serial Interface to Amplifier[7].

3.1.2 GAIN SETTING

The gains to both channels of amplifier are given as a 8-bit data, which consists of two 4-bit data. FPGA gives low AMP_CS and after that the SPI bus transfer commence. Amplifier transfer serial bit via SPI_MOSI bus at rising edge of the SPI_SCK signal[7]. MSB is first bit to be sent, means B3 bit is sent first and then followed by remaining bits for setting gain of amplifier. The gains settings values are shown in Table 3.1. Serial interface timing diagram is shown in Figure 5.

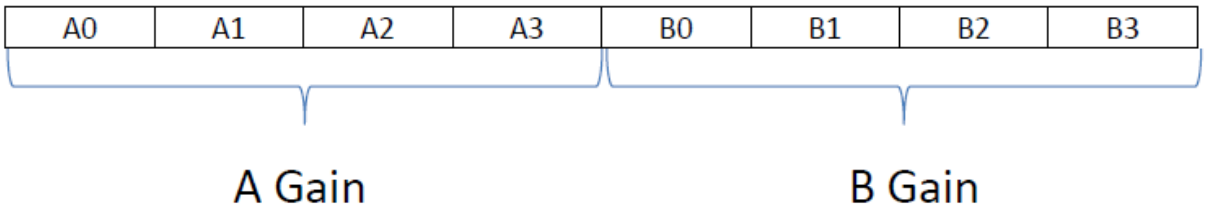

Figure 4 Gain register value given via A gain and B gain[7].

3.2 INTRODUCTION TO ADC PRESENT ON THE FPGA BOARD

LTC1407A-1 ADC device has 2 ADCs. Both the incoming inputs are sampled concurrently as soon as the AD_CONV signal goes high[7]. Some of the salient features of the onboard ADC are:

- **Sampling rate**: It use sampling rate of 3Msps. ADC device include 2 independent differential inputs. Each channel has sampling rate of 1.5 Msps.

- **3V power Supply :** Inputs supply is provided by a 3V DC.
- **Input Range:** Differential input range of the device is 2.5 V (±1.25V) gain parameters changes the input voltage range.
- **3-Wire SPI bus**: It has 3 wire serial SPI bus for communication with the FPGA. The 3wire are-AD_CONV, SPI_MISO, and SPI_SCK.

Gain	A3	A2	A1	A0	Input Voltage Range	
	B3	B2	B1	B0	Minimum	Maximum
0	0	0	0	0		
-1	0	0	0	1	0.4	2.9
-2	0	0	1	0	1.025	2.275
-5	0	0	1	1	1.4	1.9
-10	0	1	0	0	1.5875	1.7125
-20	0	1	1	0	1.625	1.675
-100	0	1	1	1	1.6375	1.6625

Table 1 Gain settings and input voltage range[7].

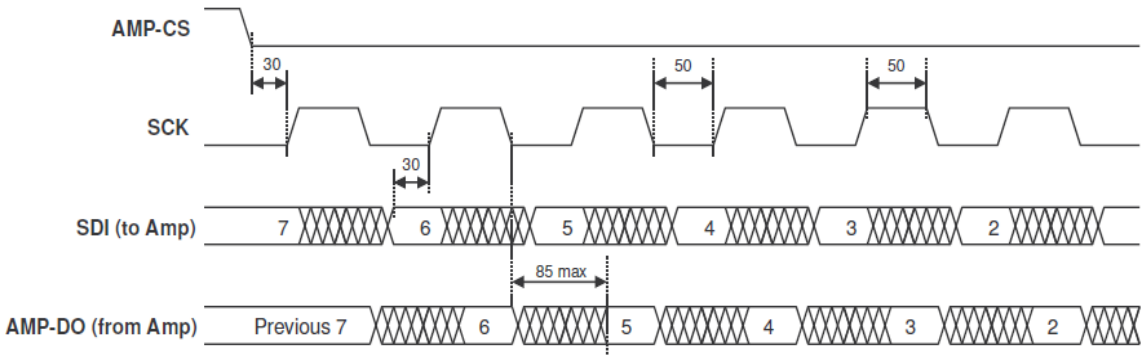

Figure 5 Timing diagram for Serial interface[8].

3.2.1 ADC INTERFACING WITH FPGA

- **AD_CONV:** The rising edge of this signal is used to trigger the sampling of the analogue inputs, to start the conversion and start the serial data transfer It is an internal signal of the FPGA board(P11). Output signal of FPGA.

- **SPI_MISO:** This signal gives serial output data from ADC device to FPGA board. It is also an internal signal(pin N10).
- **SPI_SCK:** The output of the ADC device changes as a result of the applied SPI_SCK signal rising edge.

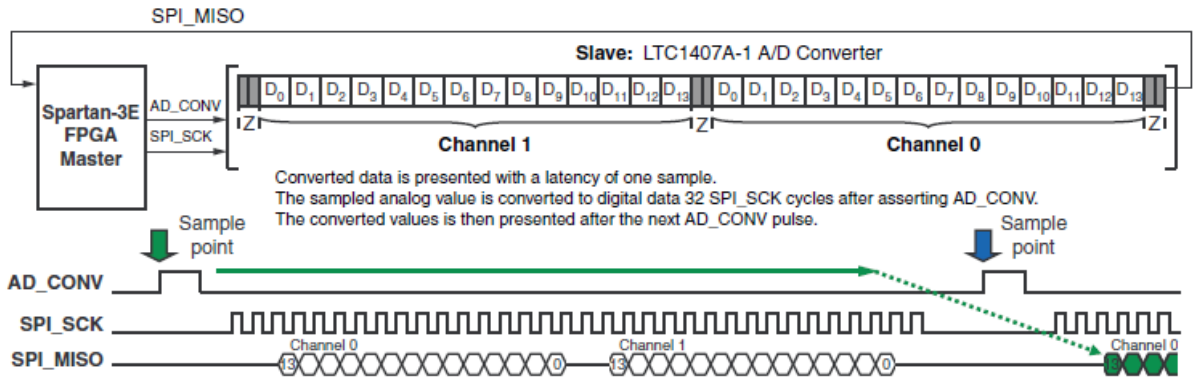

Figure 6 Timing diagram of ADC Interface with FPGA[8].

The signal conditioning circuit changed the incoming analog voltage given on VINA or VINB to 14-bit digital value[7], D[13:0], as

$$\mathbf{D[13:0] = Gain\ X\ \frac{VIN-1.65V}{1.25V}\ X\ 8192}$$

Where D[13:0] is 14 bit two's complement output from ADC device given to FPGA. Output from ADC chip is given to FPGA through the SPI_MISO bus. GAIN is the gain setting given by the AMP to the ADC chip through SPI_MOSI bus bit after bit. VIN is the input voltage to the ADC. 1.65V is reference voltage of ADC device. Reference voltage is given by voltage divider circuit in the ADC chip (dividing the VCC which is 3.3V). ±1.25V is the range of ADC used. Hence output is scaled by 1.25V. Since the output obtained is in 14 bit 2's complement form and hence the output is scaled by 8192. Both the input channels [VIN(A) and VIN(B)] are sampled concurrently.

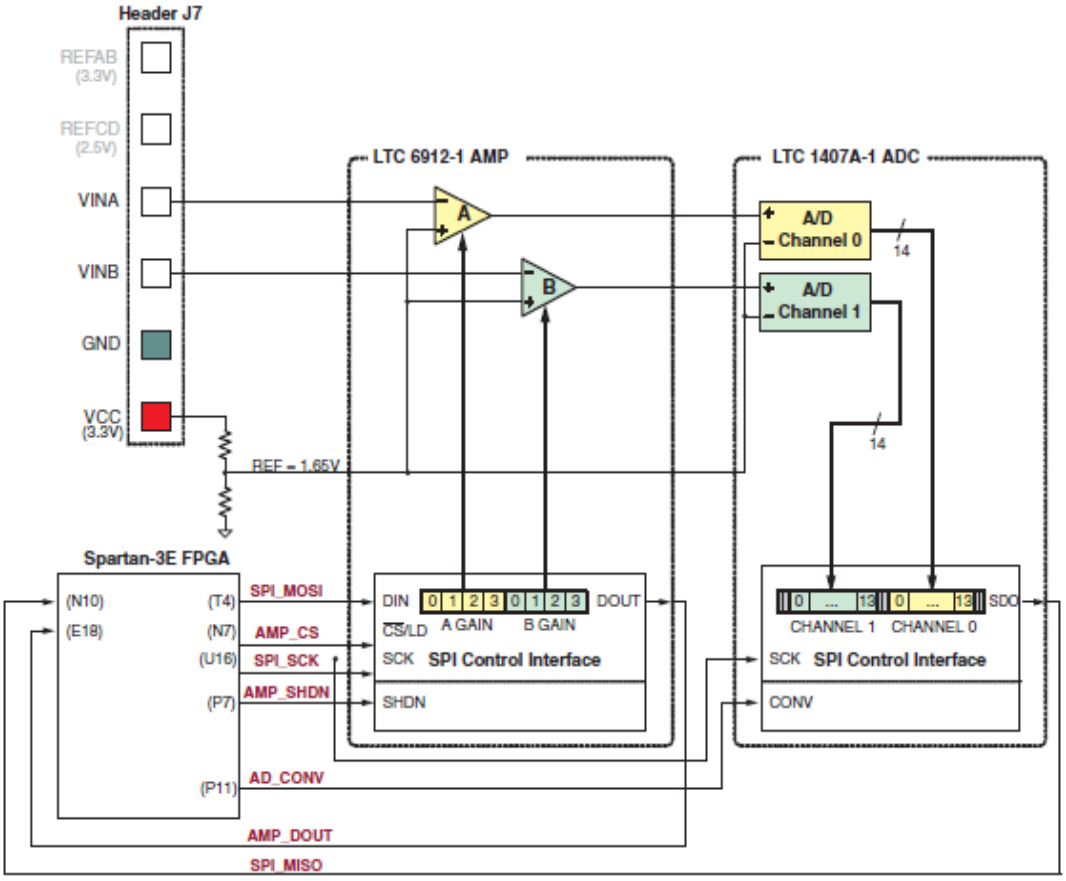

Figure 7 Internal block diagram of pre-amplifier and ADC[7].

CHAPTER 4

RS232 TRANSMITTER INTERFACING

4.1 INTRODUCTION TO RS232 SERIAL PORTS

The FPGA contain two RS-232 serial ports: a female DB9 DCE connector and a male DTE connector. DCE port connect with serial port connector which is there on computers. Whereas the DTE-style connector is used to controlled RS-232 device, like modem or printer.

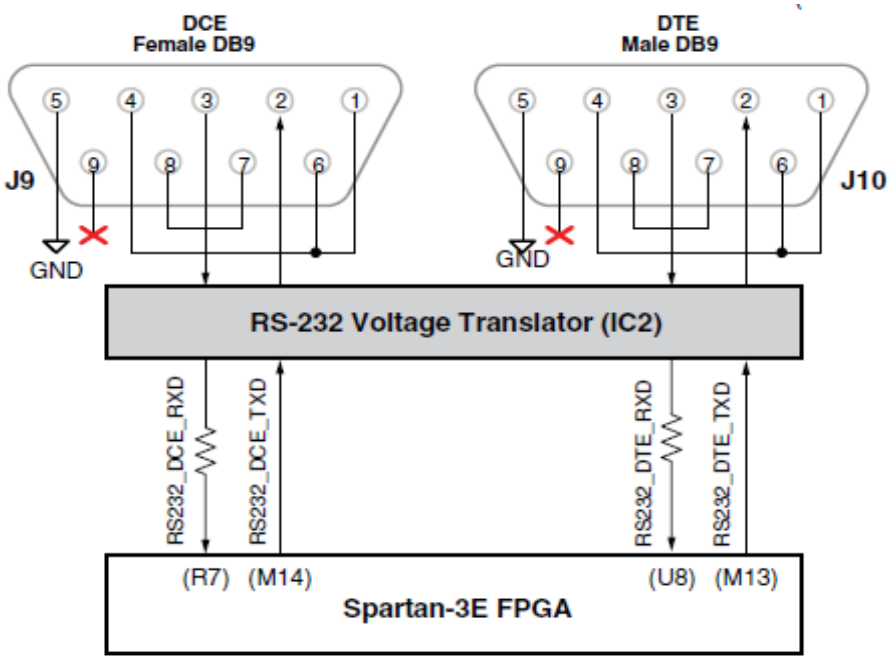

Figure 8 RS-232 Serial Ports[7]

The FPGA gives serial output data using LVTTL or LVCMOS levels to the Maxim device. The Maxim device in turn, converts the logic data to the approximate RS-232 voltage level. Also the Maxim device converts the RS-232 serial input data to LVTTL levels for the FPGA. The series resistor present between the Maxim output pin and FPGA's RXD pin save from coincidental logic interference.

4.2 MECHANISM OF TRANSMITTER INTERFACING

It is the serial communication protocol transmitting parallel data via a serial line. The transmitter is a specific shift register which can hold parallel data and then shifts it out bit after bit. Transmission normally starts once a start bit ('0') is sent, followed by the data bits

(usually 6, 7 or 8), a parity bit(optional) and finally stop bits (with 1, 1.5 or 2 '1's). The transmitter includes UART transmitter, baud rate generator and interface circuit. Figure 15 shows byte of data transmitted in serial manner.

Clock signal is not required through serial line. This demand agreement on the transmission specification from both transmitter and receiver side in advance. This information includes the band rate (number of bits per second), the number of data bits and stop bits, and whether parity is being used. The mostly used baud rates include 2400, 4800, 9600 and 19,200. The baud rate used here is 19200.

We use oversampling scheme to transmit data. This scheme uses a high frequency sampling signal (s_pulse) to transmit data. The sampling rate is 16 times the baud rate (ie., 16 sampling pulses for each bit). So the frequency of the sampling pulse is 19200X16. We use the onboard system clock which is 50MHZ. Therefore the sampling pulse should be asserted once every 162 count(50000000/19200*16).

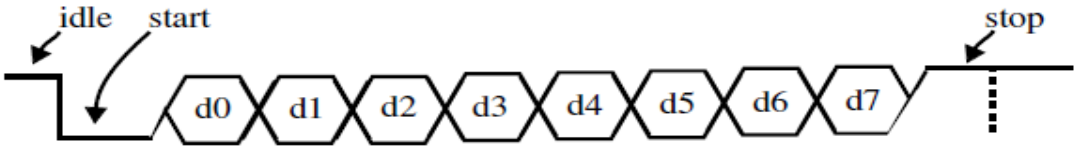

Figure 9 Serial transmission of a byte[5].

4.3 INTERFACING THE TRANSMITTER WITH FPGA

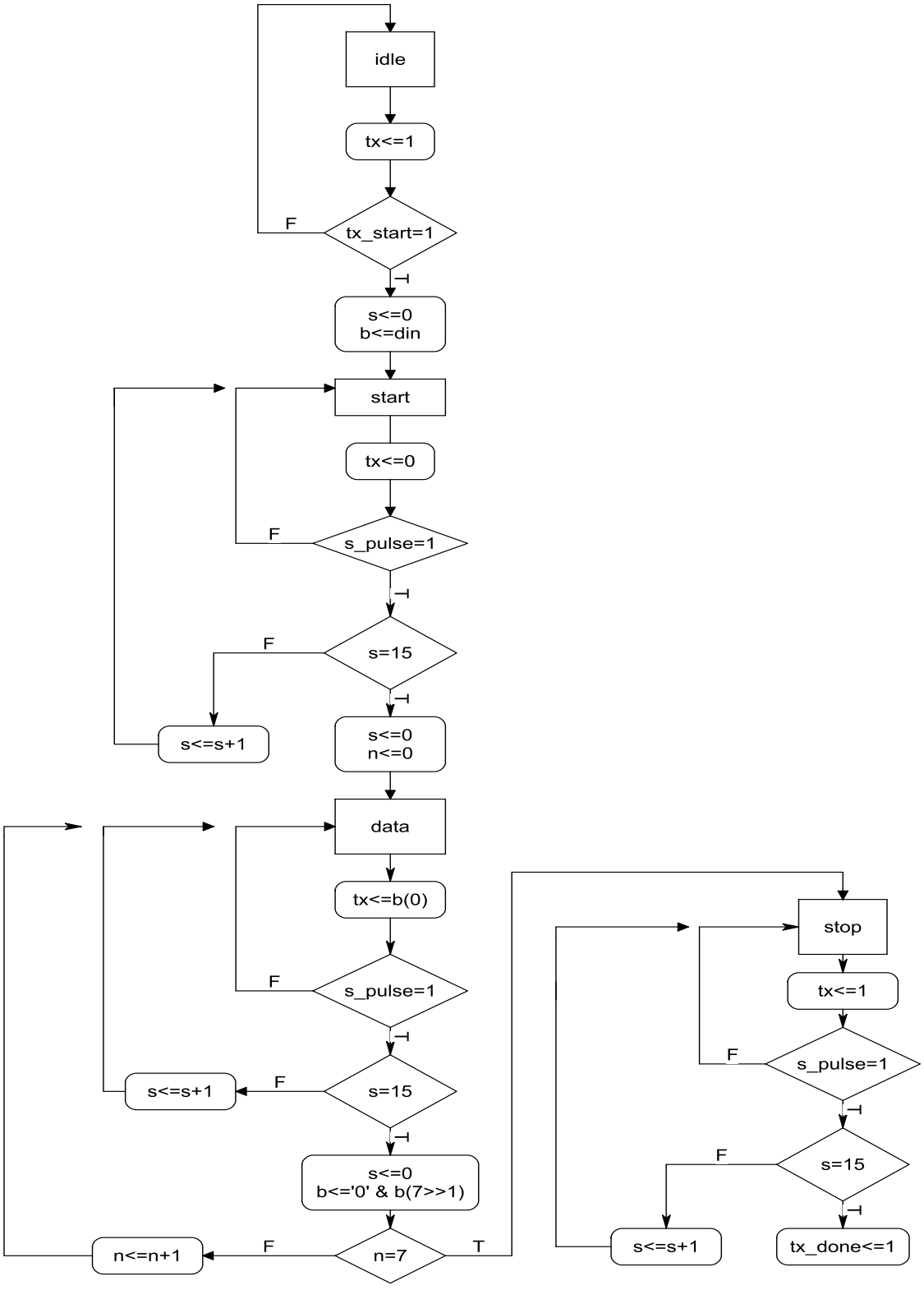

Figure 10 ASMD of the transmitter[5].

The main state machines are start, data and stop. After assertion of the tx_start signal, the FSMD loads the data word and then gradually progress through the start, data, and stop states to shift out the corresponding bits. It signals completion by asserting the tx_done signal for one clock cycle. A 1-bit buffer, tx_reg, is used to filter out any potential glitch. The s-pulse signal is the enable signal whose frequency is 16 times that of the baud rate. The complete ASMD chart for the transmitter is shown in figure 16.

CHAPTER 5

LCD INTERFACING

5.1 INTRODUCTION TO LCD

The Spartan-3E FPGA board includes 2-line of 16-character LCD. 4-bit data interface method is used by FPGA to control the LCD. This LCD display also support 8-bit data interfacing, but still the FPGA board use only 4-bit data interface so that it can minimize the total pin count and also be compatible with other Xilinx development board.

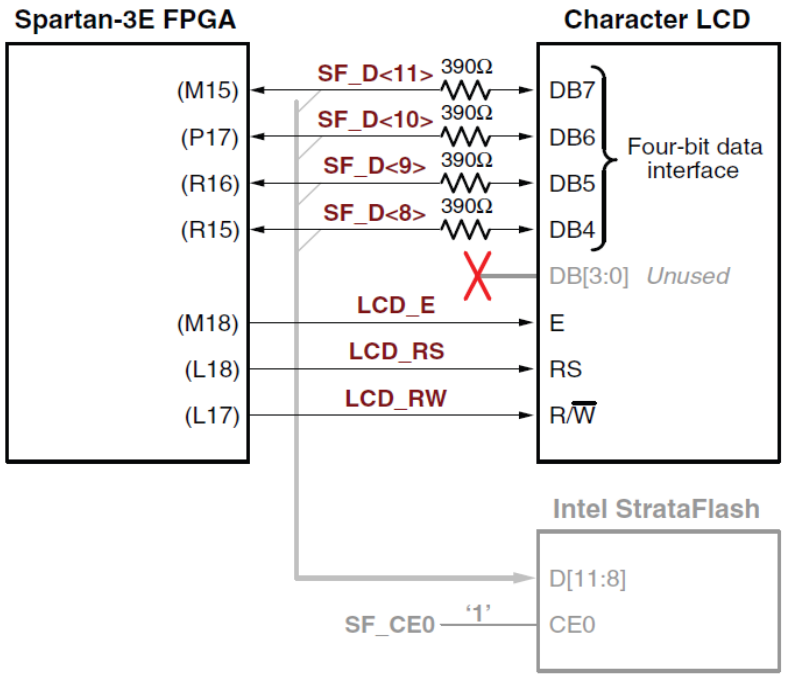

Figure 11 LCD interface detailed[7].

Interfacing to and controlling LCD device is not complex problem. The LCD module can be controlled by sending serial bytes of data in ASCII format. Graphic display requires sequence of command bytes followed by bytes of data. The command bytes control various modes of operation like clear display, set display, return cursor home, cursor shift, function set among others. Table 3 shows the different pin configuration for interfacing LCD module with the FPGA. The display is not fast.

Signal Name	FPGA Pin	Function	
SF_D<11>	M15	Data bit DB7	Shared with StrataFlash pins SF_D<11:8>
SF_D<10>	P17	Data bit DB6	
SF_D<9>	R16	Data bit DB5	
SF_D<8>	R15	Data bit DB4	
LCD_E	M18	Read/Write Enable Pulse 0: Disabled 1: Read/Write operation enabled	
LCD_RS	L18	Register Select 0: Instruction register during write operations. Busy Flash during read operations 1: Data for read or write operations	
LCD_RW	L17	Read/Write Control 0: WRITE, LCD accepts data 1: READ, LCD presents data	

6. 1 LCD Various Pin configuration[7]

Function	LCD_RS	LCD_RW	Upper Nibble				Lower Nibble			
			DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Clear Display	0	0	0	0	0	0	0	0	0	1
Return Cursor Home	0	0	0	0	0	0	0	0	1	-
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S
Display On/Off	0	0	0	0	0	0	1	D	C	B
Cursor and Display Shift	0	0	0	0	0	1	S/C	R/L	-	-
Function Set	0	0	0	0	1	0	1	0	-	-
Set CG RAM Address	0	0	0	1	A5	A4	A3	A2	A1	A0
Set DD RAM Address	0	0	1	A6	A5	A4	A3	A2	A1	A0
Read Busy Flag and Address	0	1	BF	A6	A5	A4	A3	A2	A1	A0
Write Data to CG RAM or DD RAM	1	0	D7	D6	D5	D4	D3	D2	D1	D0
Read Data from CG RAM or DD RAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0

Figure 12 Command set for the LCD display[7].

5.2 INTERFACING LCD WITH FPGA

Every byte of command provided to LCD interfacing occur using 4 bit interfacing, hence, each command is divided into two 4 bit transmissions spaced by 1us. Successive commands (all sequential 4 bit transmission) need to seperate from next for 40us.

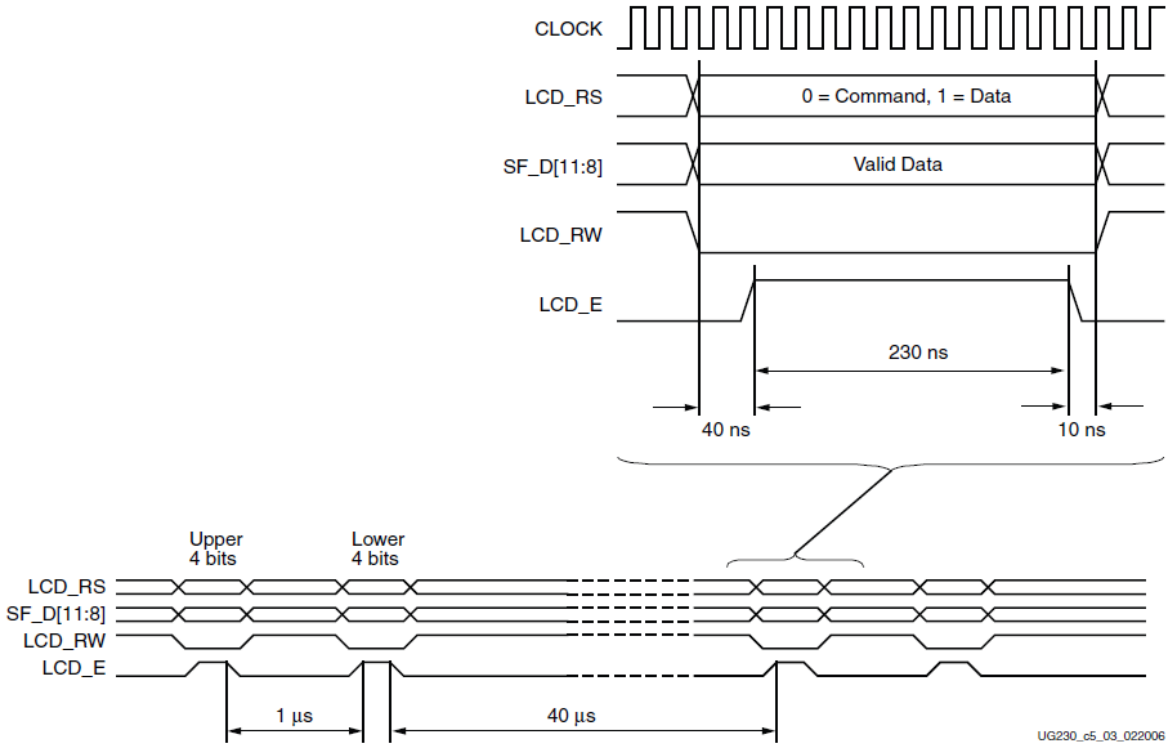

Figure 13 Timing diagram for the LCD interface[7].

The 3 major process in interfacing the LCD are initialization of 4 bit interfacing, the second is to set commands for displaying operation and 3rd is writing of character.

One design machine is for power on initialization process, one state machine is for transmitting commands and data byte to LCD module and the other one is to commence the power on initialization process.

5.2.1 Initialization

(a) First wait for 15 ms or may be more, even though display is set after FPGA is done with configuration.

(b) Then SF_D<11:8> = 0x3, the signal LCD_E goes high by twelve cycle of clk.

(c) Then wait for another 4.1 ms or more.

(d) Now SF_D<11:8> = 0x3, LCD_E again rise by twelve clk cycles.

(e) Again wait for another 100 µs or more.

(f) Again SF_D<11:8> = 0x3.

(g) Wait again for 40 µs.

(h) Now SF_D<11:8> = 0x2, LCD_E rise by twelve clk cycles.

(i) Finally wait for 40 µs.

The second step involves the configuration and actual writing to the LCD ram.

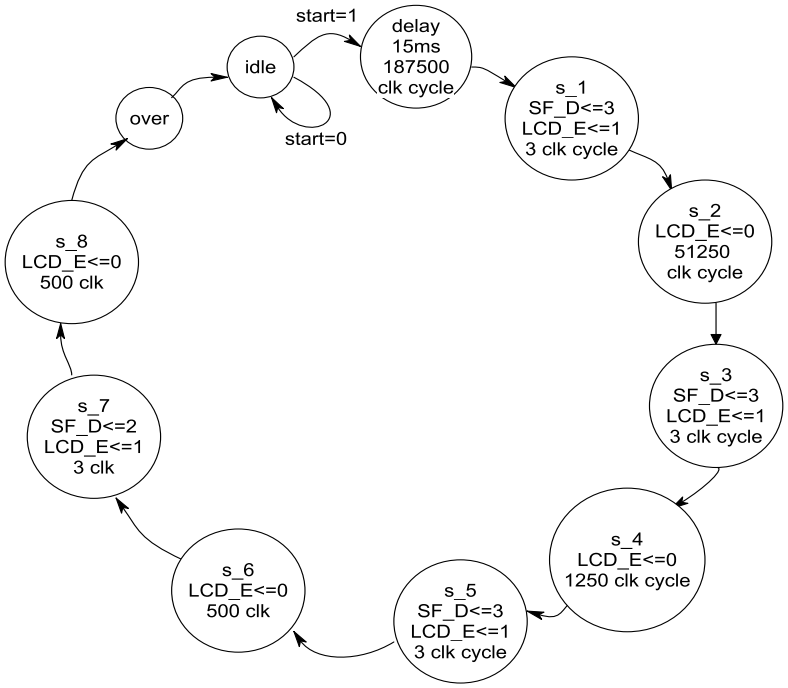

Figure 14 Initialization[7].

5.2.2 Configuration

1. First insert 0x28 to set Function for configuring display.

2. Insert 0x06 to set Entry Mode for setting display for automatically incrementing address pointer.

3. Insert 0x0C to On/Off Display for turning display on and also disable cursor

and also blinking.

4. Lastly, Clear the Display

The final step includes actual processing of data writing to DD-RAM.

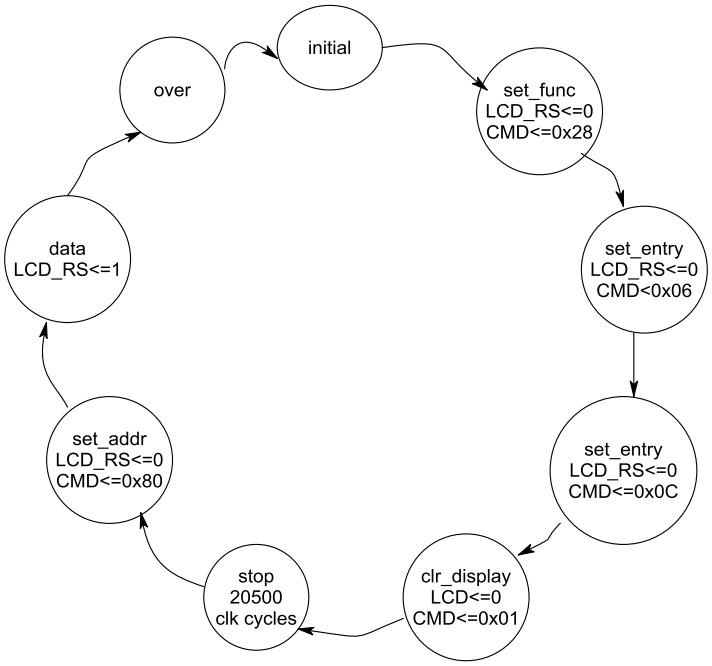

Figure 15 configuration[7].

5.2.3 Display

1. Specify the start address with a Set DD-RAM Address command.

2. Display a character with a Write Data command.

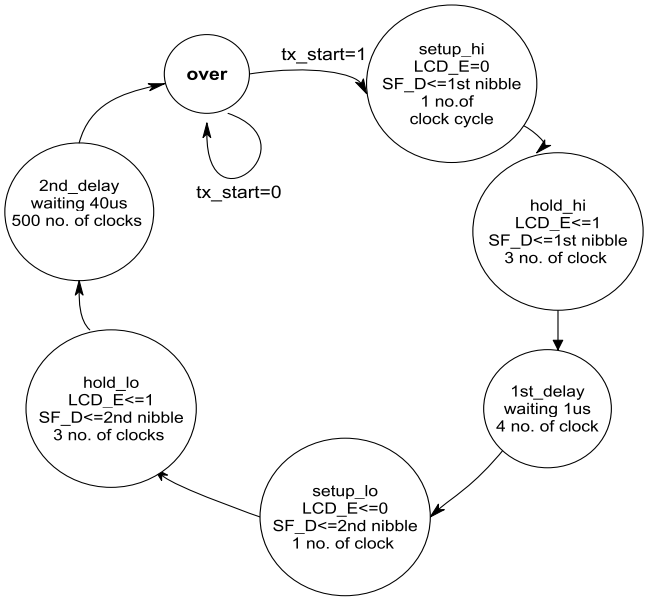

Figure 16 display state machine[7].

CHAPTER 6

SIMULATION RESULTS

6.1 STEP BY STEP RESULTS

Step by step result of each designed module is given below. The dc signal voltage applied to the onboard A/D converter after conversion is obtained as 2's complement 14 bit data. The output from the ADC is finally transmitted serially using RS232 transmitter. It is also display on the LCD screen for diagnosis if the received data is valid.

6.2 SIMULATION RESULT OF PRE-AMPLIFIER & ADC INTERFACE

The design is to control Linear Technology LTC6912-1 programmable preamplifier and Linear Technology LTC1407A-1 ADC onboard the Spartan 3E starter kit FPGA board.

6.2.1 SIMULATION WAVEFORM

The design is simulated, synthesised and implemented on the FPGA board using the Xilinx ISE 13.2 simulator tool. VHDL is used to program the FPGA board via USB on the PC. The design uses 64 states, 2 inputs, 10 outputs, clock, and reset.

Logic Utilization	Used	Available	Utilization
Number of slices	48	4656	1%
Number of Slice Flip Flops	64	9312	0%
Number of 4 input LUTs	47	9312	0%
Number of bonded IOBs	31	232	13%
Number of GCLKs	1	24	4%

Table 2 Device Utilization Summary of the ADC interfacing

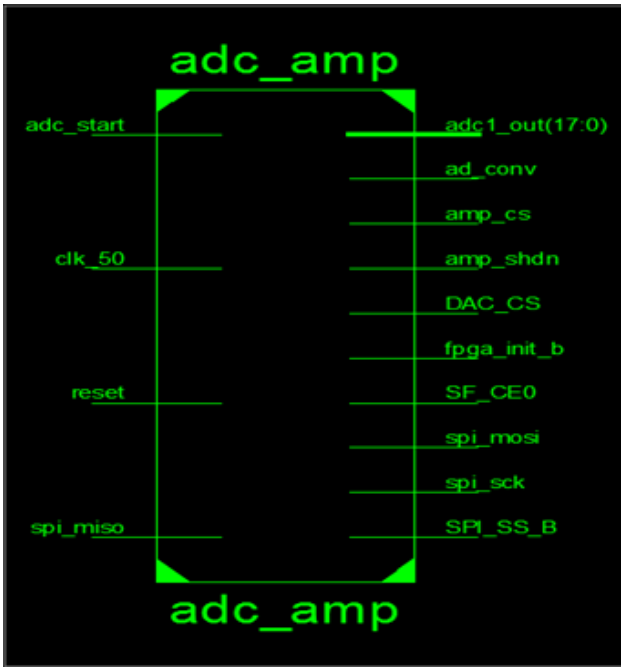

Figure 17 Top level schematic of the ADC interfacing

The test bench waveform of the ADC controller is shown in figure 25. As can be seen from the figure the AMP_CS signal goes low before the gain setting bit is given to the AMP by the

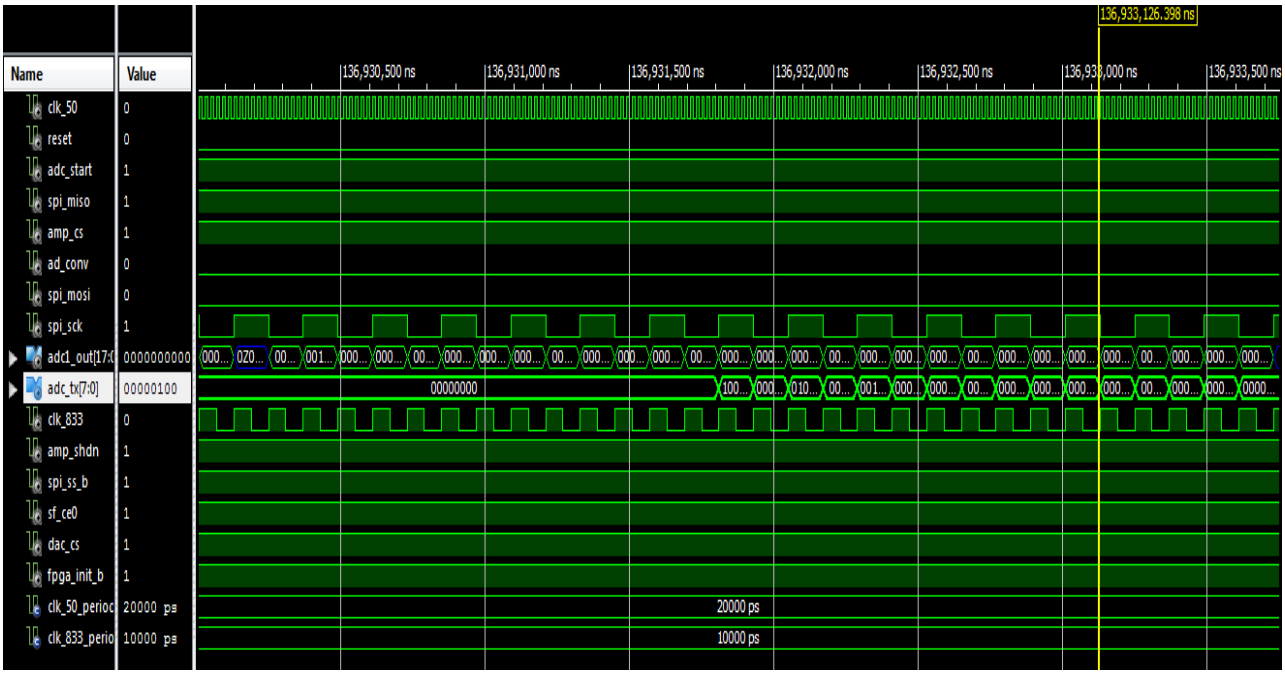

Figure 18 Test bench waveform of the ADC interfacing

FPGA. AMP_CS signal again goes high to finish gain setting. AD_CONV signal is asserted 3ns before conversion takes place. Total number of SPI_SCK clock cycles used for conversion is

18. In the first two clock cycles and the last two cycles SPI_MISO is driven to high impedance. The output bit is transfer to the FPGA from the ADC via SPI_MISO signal at the rising edge of the SPI_SCK clock from the second clock. The 14 bit ADC output is obtained in 2's complement form.

6.3 SIMULATION RESULT OF RS232 TRANSMITTER INTERFACING

Rs232 transmitter sent one byte data serially to the GSM module for transmission or the Hyper Terminal window for diagnosis.

6.3.1 SIMULATION WAVEFORM

The design is simulated and synthesis in Xilinx ISE 13.2 simulator tool. The design use 4 states, 2 inputs, 2 outputs, clock and reset.

Logic Utilization	Used	Available	Utilization
Number of slices	47	4656	1%
Number of Slice Flip Flops	26	9312	0%
Number of 4 input LUTs	88	9312	0%
Number of bonded IOBs	13	232	5%
Number of GCLKs	1	24	4%

Table 3 Device Utilization Summary of transmitter interfacing

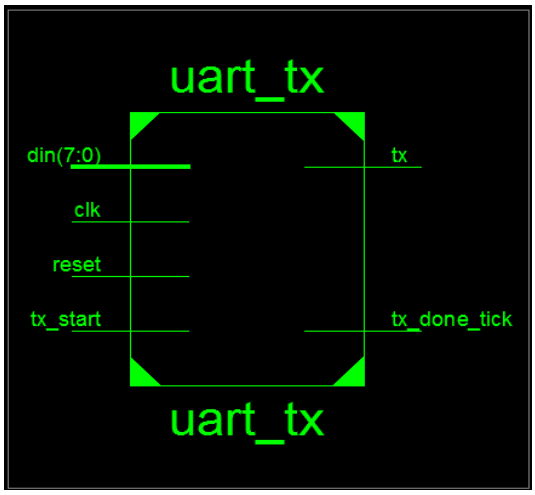

Figure 19 Top-level schematic of the Transmitter interface

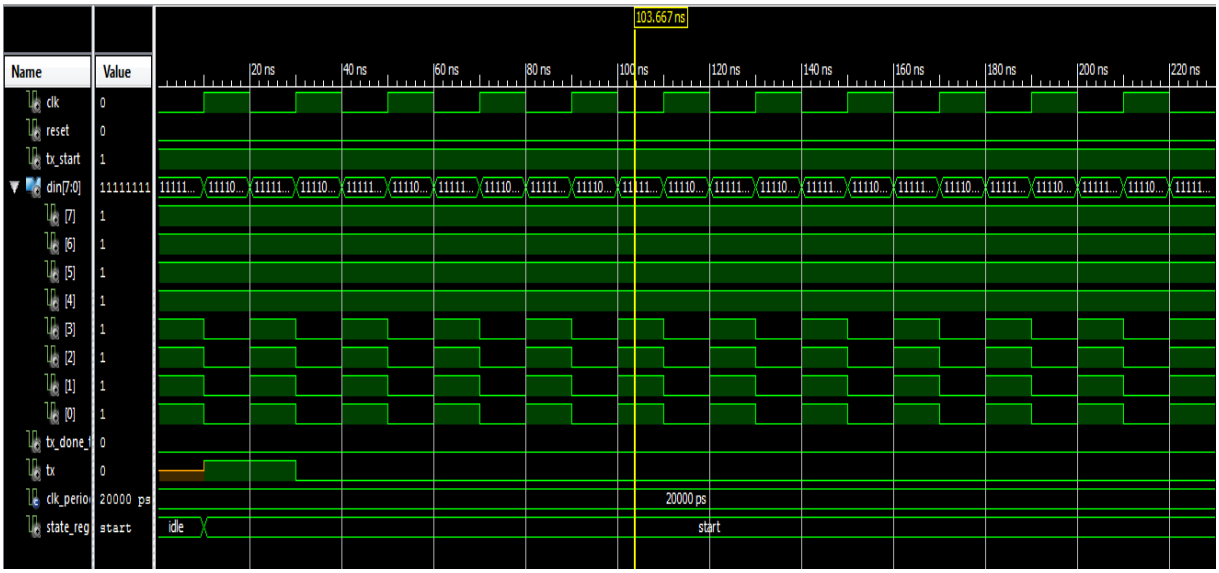

Figure 20 Simulation waveform for the transmitter interface

The transmitter output can be given to GSM wireless module for further transmission to control room. Output is obtained in Hyper Terminal window for diagnosis to check if the transmitted data is valid one.

Figure 21 Transmitter output in Hyper Terminal

6.4 SIMULATION RESULT OF LCD INTERFACING

The output from the memory is displayed on the LCD for testing/diagnosis to check if the data received on the LCD is valid.

6.4.1 SIMULATION WAVEFORM

The design use 27 states machine, 1 input, 6 outputs, clock and reset. The controller is simulated and synthesis in Xilinx ISE 13.2 simulator tool by Xilinx. The onboard 50MHZ clock is divided to get 12.5MHZ.

Logic Utilization	Used	Available	Utilization
Number of slices	48	4656	1%
Number of Slice Flip Flops	64	9312	0%
Number of 4 input LUTs	47	9312	0%
Number of bonded IOBs	31	232	13%
Number of GCLKs	1	24	4%

Table 4 Device Utilization Summary of the LCD interfacing

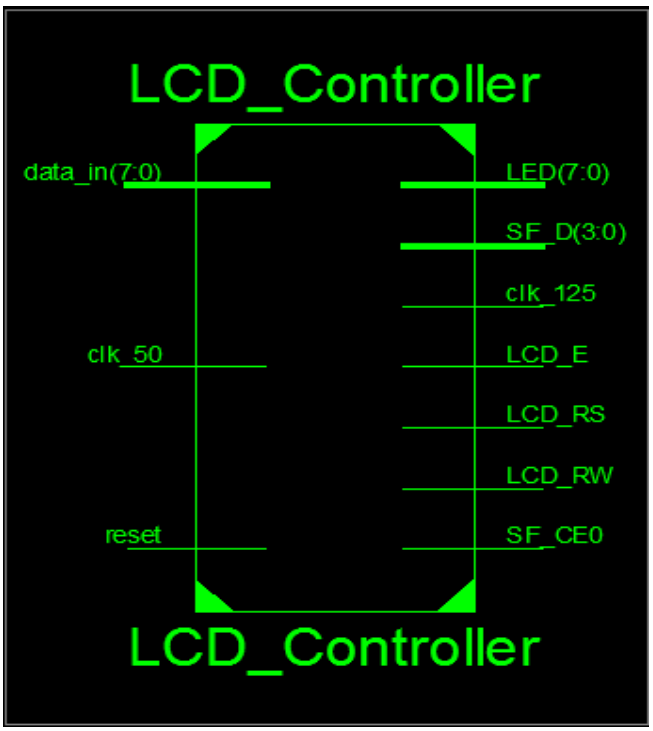

Figure 22 top level diagram of LCD interface

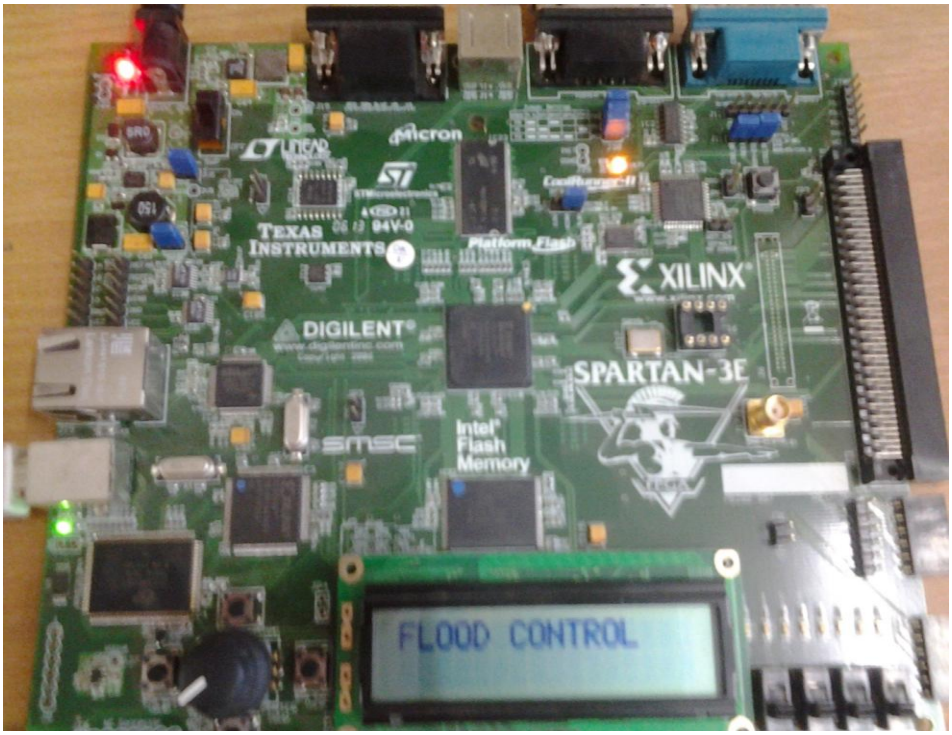

Figure 23 Output display on the LCD

Besides transmitting the data via RS232, the character is also display on the LCD to verify if the data displayed is a valid data as expected.

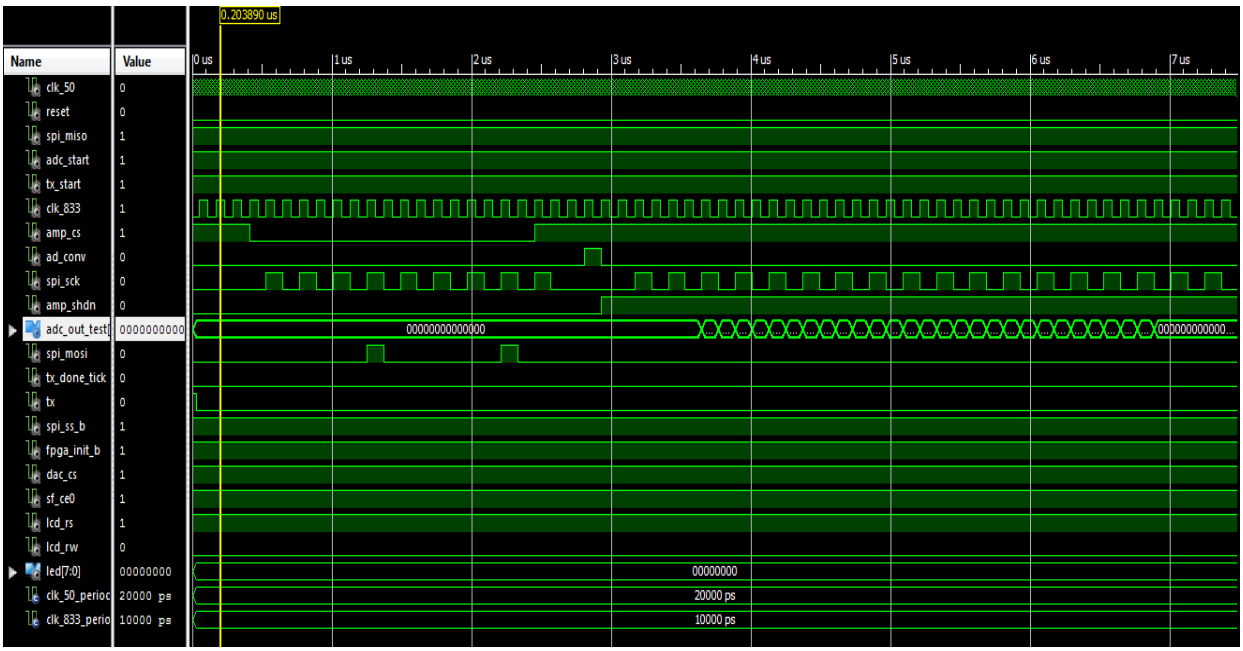

Figure 24 Simulation waveform for the complete system

CHAPTER 7

CONCLUSION & SCOPE FOR FUTURE WORK

7.1 CONCLUSION

In this project embedded system on flood monitoring is design and implemented in hardware on Spartan 3E Starter kit FPGA board. The system is implemented and tested in the laboratory. The output result is obtained In Hyper Terminal window and also display on the LCD. This design can be of great help in real life in the flood prone area especially near the river and dam.

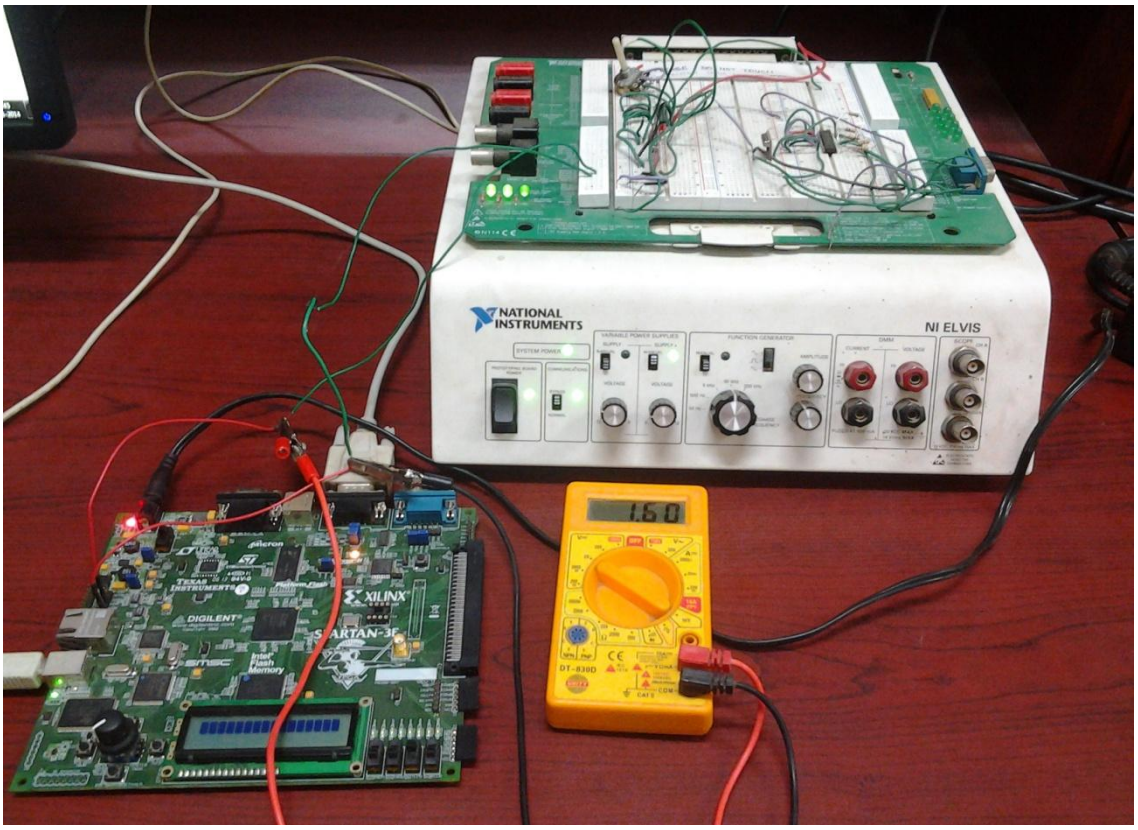

Figure 25 Hardware implementation of the system design.

7.2 SCOPE FOR FUTURE WORK

For the most effective use of the system, multiple numbers of sensors can be connected to the board using a multiplexer so the water level can be read more accurately.

Also GSM/GPRS module can be used to transmit the data via wireless to the control room.

BIBLIOGRAPHY

[1] Sandeep Gandla, Waleed K. Al-Assadi, Sahra Sedigh and Raghu A. R. Rao(2008), Design and FPGA Prototyping of a Flood Prediction System, Department of Electrical and Computer Engineering, Missouri University of Science and Technology.

[2] Inyiama H. C., Obota M. E.,(2013), Designing Flood Control Systems Using Wireless Sensor Networks, International Journal of Engineering Research and Applications (IJERA), Vol. 3, Issue 1, January -February 2013, pp.1374-1382.

[3] Jirapon Sunkpho, Chaiwat Ootamakorn,(2011), Real-time flood monitoring and warning system, Songklanakarin Journal of Science and Technology, 33 (2), 227-235, Mar. - Apr. 2011.

[4] Xilinx LogiCORE IP external memory controller datasheet.

[5] Chu P. Pong(2008), fpga prototyping by vhdl examples, a john wiley & sons, inc.

[6] Pedroni, Volnei A.(2004), digital electronics and design with vhdl.

[7] Xilinx, "Spartan-3E FPGA starter kit Board User Guide".

[8] K. Chapman(2006), "Picoblaze: amplifier and A/D Converter Control for Spartan-3E Starter Kit", Xilinx Ltd. 2006.

[9] Linear Technology Limited, Datasheet of LTC 1407A-1.

[10] Linear Technology Limited, Datasheet of LTC 6912-1.